Time Series Forecasting of COVID-19 Cases Using ARIMA and Prophet Models

Abstract

This technical report explores the forecasting of COVID-19 confirmed cases in India using two prominent time series modeling techniques: ARIMA (AutoRegressive Integrated Moving Average) and Prophet (developed by Facebook/Meta). The dataset utilized originates from Johns Hopkins CSSE and consists of daily cumulative case counts. The project encompasses complete steps from data extraction and transformation to model building, forecasting, and error evaluation. Results are interpreted thoroughly to compare model effectiveness and suitability for different forecasting scenarios.

1. Introduction

The COVID-19 pandemic necessitated the application of reliable forecasting tools to assist in policymaking and healthcare resource planning. Time series forecasting is a fundamental approach to predict future values based on historical patterns. Two widely adopted methods are ARIMA and Prophet. ARIMA is grounded in statistical principles and is effective for stationary data, while Prophet is an additive model particularly suited for data with strong seasonal effects and trend shifts. This study presents a comparative analysis of both techniques using real-world COVID-19 case data.

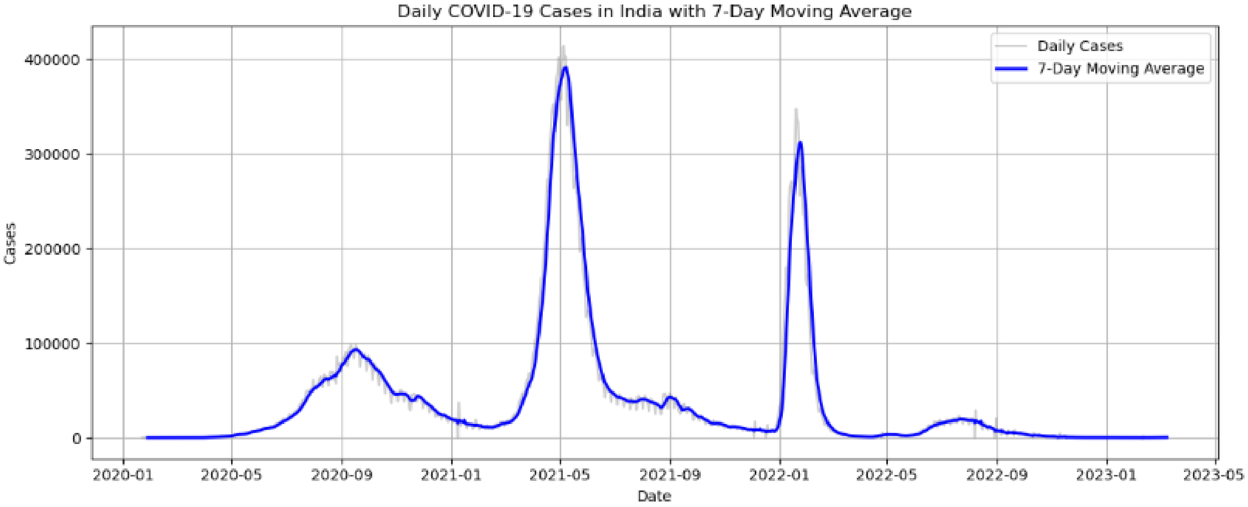
2. Dataset Description and Preprocessing

The dataset used in this study was obtained from the publicly available Johns Hopkins University CSSE COVID-19 repository. It contains global cumulative confirmed COVID-19 case counts recorded daily from January 22, 2020, through March 9, 2023. Each row in the dataset represents a geographical entity such as a country or region, and the columns represent the number of confirmed cases on each day. The dataset is structured with metadata fields such as 'Province/State', 'Country/Region', 'Lat', and 'Long', followed by over 1100 columns representing consecutive dates.

For this project, only the rows corresponding to 'India' were retained. The cumulative confirmed cases were aggregated across all states and union territories. Non-numeric columns including geographical metadata were removed, and only the date-wise case counts were retained. The column headers were converted to datetime objects using pandas' to_datetime with error handling to ensure consistent formatting. Since the dataset records cumulative cases, the daily new cases were computed by differencing consecutive values using the .diff() function and replacing the initial NaN with 0. This transformation is a prerequisite for stationary modeling, particularly for ARIMA.

3. Exploratory Data Analysis (EDA)

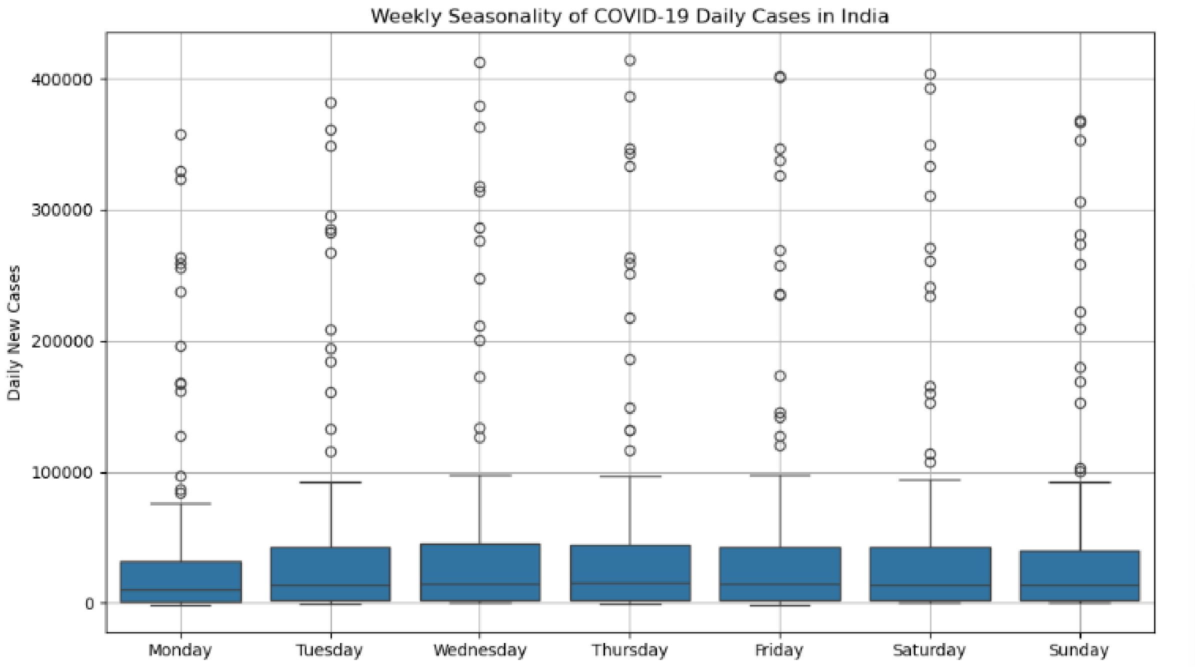
Exploratory analysis involved visualizing the daily case trends using line plots and computing a 7-day moving average to smooth out daily fluctuations. This moving average helped highlight underlying trends by dampening day-to-day noise. Further, box plots categorized by the day of the week were generated to explore potential weekly seasonality. This revealed periodic dips on weekends, consistent with known patterns of reduced testing or reporting.



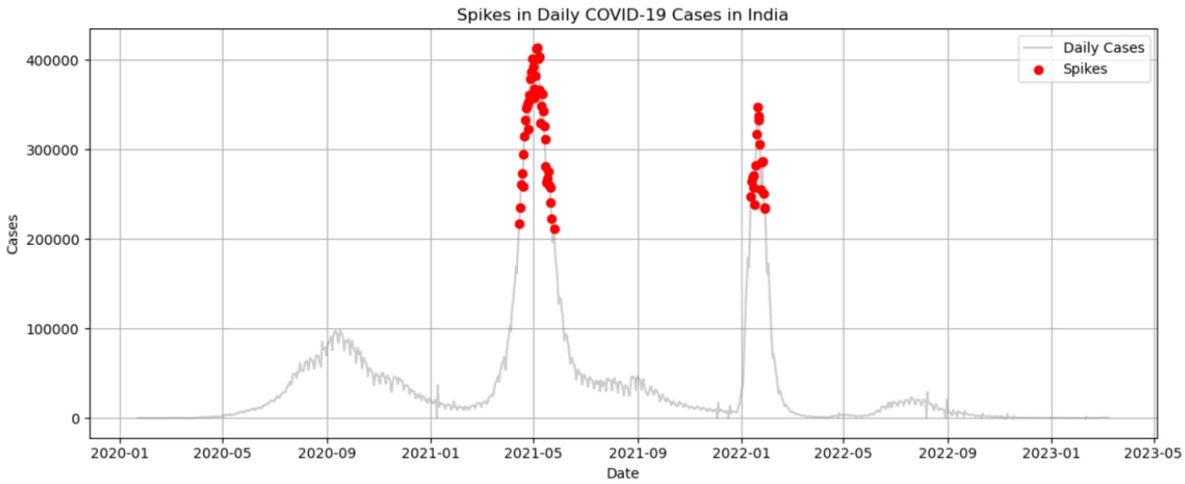
To explore weekly seasonality in COVID-19 case reporting, a box plot was generated to visualize the distribution of daily new cases for each day of the week. The box plot reveals that while the median daily cases remain somewhat consistent across the week, there is noticeable variability in the spread and frequency of outliers.

Interestingly, Mondays show a slight dip in the median and interquartile range compared to midweek days like Wednesday and Thursday. This supports the known phenomenon of reduced case reporting on weekends, leading to lower numbers on Mondays. Conversely, case counts tend to rebound during midweek as reporting catches up. Additionally, the presence of extreme outliers across all days indicates high variability due to pandemic waves and data backlogs.

This analysis confirms that weekly seasonality, though not dominant, exists to some extent in the dataset and can influence the performance of models like Prophet that factor in such cyclical behavior. Understanding this reporting bias helps in fine-tuning forecasts and interpreting short-term fluctuations more accurately.



In addition, spikes in case counts were detected using a 95th percentile threshold, visually represented to identify outlier days possibly corresponding to pandemic waves. Time series decomposition using the seasonal decomposition function provided separate views of trend, seasonality, and residual components. These analyses set the foundation for choosing appropriate forecasting methods.

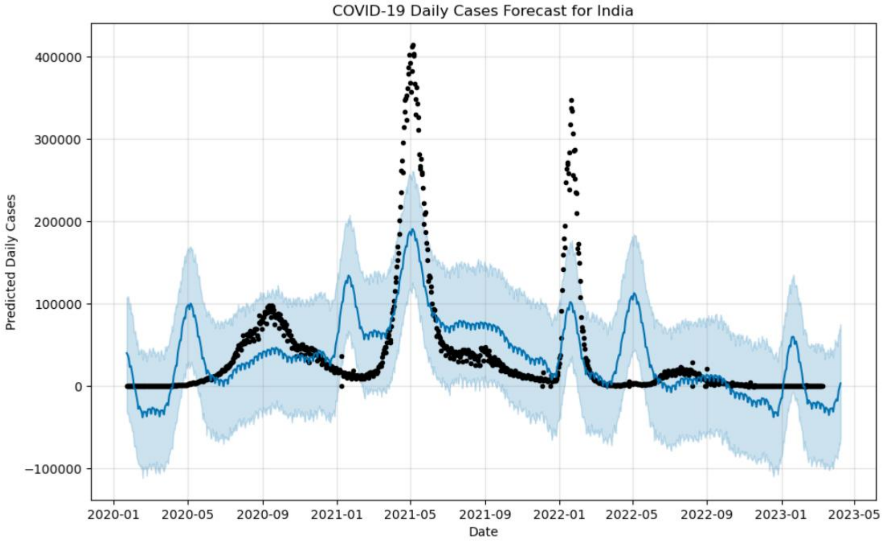


4. Forecasting Using Prophet

The Prophet model was trained on data up to January 31st , 2022, and configured to forecast the subsequent 30 days. Prophet handles trends and seasonality through an additive model framework, decomposing the time series into trend, weekly, and yearly seasonal components. It does not require the data to be stationary, making it accessible for users without extensive time series background.

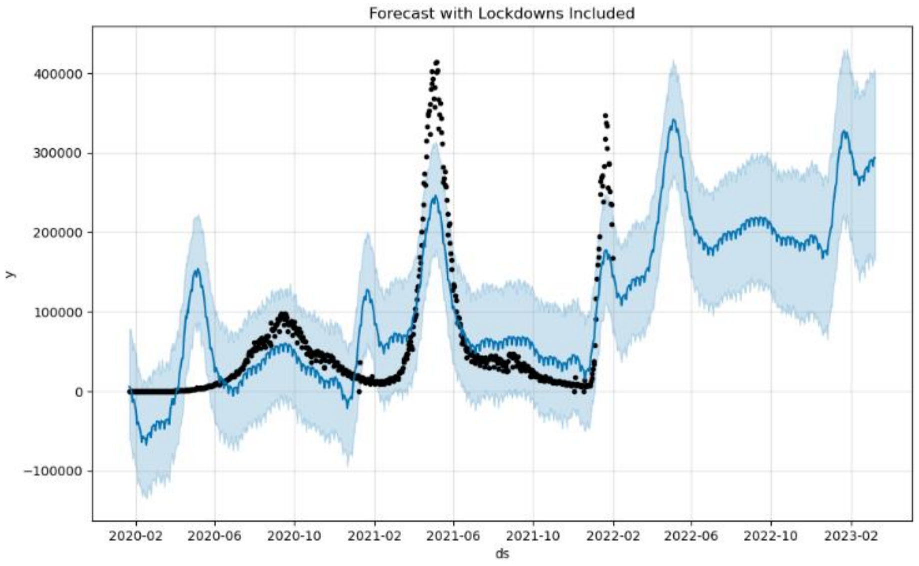
Implementation involved initializing the model and fitting it to the training dataset. Prophet's make_future_dataframe method generated the future dates for prediction. The model produced a forecast with predicted values (yhat) and confidence intervals (yhat_lower and yhat_upper). The output was visualized with Prophet's built-in plotting tools, showing how well the model captured the data's components.

However, when evaluated against actual case values, Prophet yielded a mean absolute error (MAE) of 37770.96 and a root mean squared error (RMSE) of 50429.40. This relatively high error indicated overestimation, likely due to Prophet projecting future growth based on historical upward trends without considering the recent flattening in reported cases.



When Prophet was initially applied to this dataset, the forecast results were suboptimal due to the model's reliance on detecting regular seasonal patterns and trend-based extrapolation. COVID-19 case data, particularly during 2021–2023, was heavily influenced by irregular events such as lockdowns, vaccination drives, and policy shifts. These real-world interventions caused sudden, non-cyclical changes in case numbers that Prophet could not capture using its default assumptions.

To address this, the model was enhanced by introducing lockdown dates as 'holidays' in Prophet. Prophet's holiday feature allows the user to specify dates that significantly influence the target variable, enabling the model to learn temporary shocks or dips in the trend. This modification improves the model's adaptability to real-life disruptions, resulting in a more realistic and context-aware forecast of case trends.



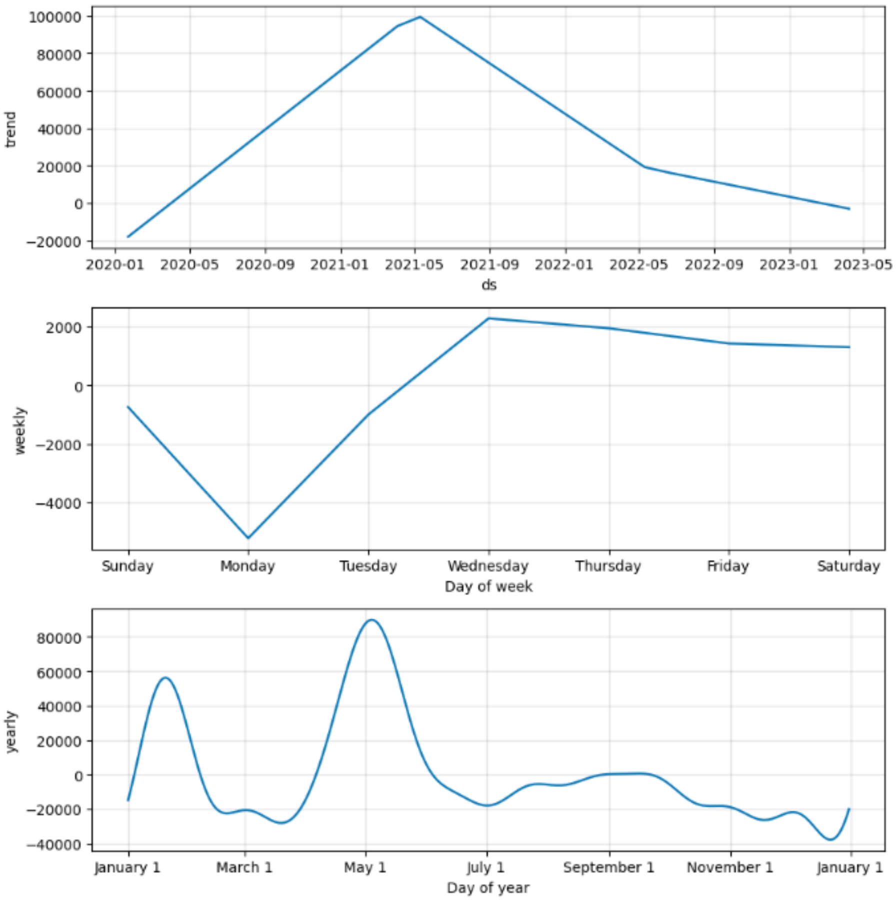
After Prophet generates a forecast, it provides a breakdown of the time series into interpretable components. The figure below represents the output of Prophet's plot_components() function, which visualizes how each part contributes to the forecast.

The first subplot shows the overall trend. It captures the long-term progression of COVID-19 cases in India. From this plot, we observe a strong upward trend that peaks around mid-2021, which aligns with the timing of the Delta variant wave. Following the peak, the trend declines steadily through 2022 and remains relatively low into early 2023. This indicates that Prophet effectively learned the pandemic wave patterns from historical data.

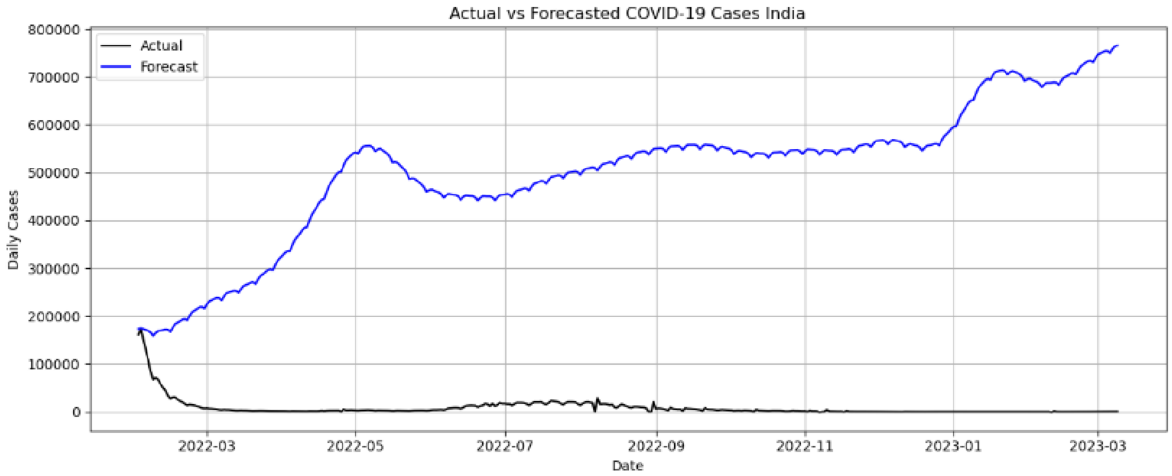
The second subplot displays the weekly seasonality component. It reveals recurring weekly patterns in reported cases. The graph indicates a noticeable dip on Mondays, which corresponds to the well-documented reporting lag caused by fewer tests conducted on weekends. A gradual rise is observed from Tuesday through Friday, followed by a decline into the weekend. This insight is crucial for understanding daily case fluctuations unrelated to disease spread but rather reporting artifacts.

The third subplot illustrates the yearly seasonality, highlighting cyclic patterns across the calendar year. In this case, the plot suggests that spikes tend to occur around late March and early May, which correspond to the timing of previous pandemic waves in India. The rest of the year shows a relatively low and fluctuating pattern. Prophet captures this using Fourier series to model periodic fluctuations over a year.

Together, these components allow the Prophet model to adjust forecasts dynamically based on structural patterns in the data. However, the accuracy of such decomposition depends on the volume and quality of input data, as well as the presence of genuine seasonal effects—which in the case of COVID-19, may vary depending on public health responses and variant outbreaks.



One of the key evaluation plots generated in the notebook shows the Prophet model's forecast (in blue) against the actual recorded cases (in black). In this visualization, the model consistently overpredicts daily case numbers beyond early 2022. The forecast curve exhibits a strong upward trend that does not align with the relatively flat trajectory of the actual data. This discrepancy highlights the model's inability to adjust automatically to post-peak flattening in the absence of seasonal cues. Despite the addition of lockdown information, Prophet still tends to extend historical trends forward unless explicitly informed about structural breaks or contextual anomalies. This reinforces the need to supplement the model with real-world knowledge to improve its forecasting credibility.

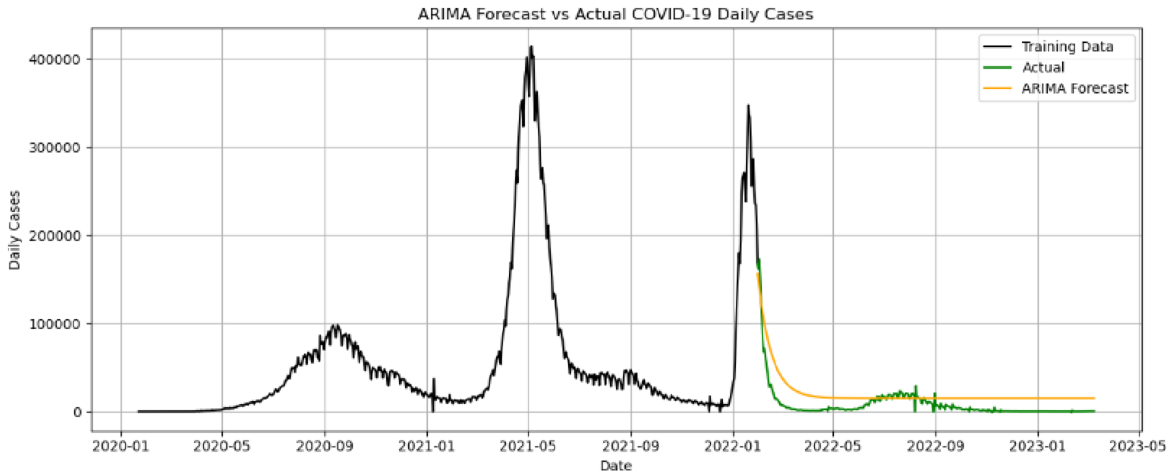


5. Forecasting Using ARIMA

The ARIMA model operates on stationary time series, so the daily case data was differenced to meet this assumption. A model with parameters ARIMA(2,1,2) was chosen based on manual tuning. The ARIMA framework consists of three terms: the autoregressive (AR) part models the dependence on lagged observations, the integrated (I) part accounts for differencing to achieve stationarity, and the moving average (MA) part models the dependency on past forecast errors.

The model was fitted on the same training dataset and used to forecast the same period as Prophet. Predictions were then evaluated against the actual test set. The ARIMA model achieved a MAE of 13550.68 and RMSE of 15693.16, significantly outperforming Prophet. This suggests that ARIMA was better suited for the post-peak plateaued nature of COVID-19 case reporting in India.

The forecasting plot visually confirmed that ARIMA followed the actual trend more closely than Prophet, which had generated phantom peaks due to assumed trend continuation.



6. Result Interpretation and Discussion

The analysis revealed that the ARIMA model significantly outperformed Prophet in forecasting COVID-19 daily cases for the chosen time period. This superior performance is attributed to ARIMA’s statistical design, which excels in handling stationary data and noise-dominated series without assuming continued growth or repetitive patterns. In contrast, Prophet is optimized for datasets with clear seasonal and trend components. However, in this case, the data following 2022 showed a flattening trend with reduced seasonality, making Prophet’s structural assumptions less suitable.

Prophet's framework attempts to extrapolate historical trends and periodic behavior into the future. When the recent data does not exhibit such characteristics, the model tends to overfit or project unrealistic surges, as seen in this study. ARIMA, by relying only on recent values and their differences, adapts better to data without strong seasonal patterns.

Nonetheless, Prophet's performance can be improved by incorporating domain knowledge in the form of external regressors. For instance, lockdowns, vaccination rollouts, and public holidays—events known to affect case reporting can be added as holidays in Prophet’s configuration. This allows the model to adjust forecasts around those dates, potentially making it more responsive to real-world interventions.

7. Conclusion

This study demonstrates a complete pipeline for time series forecasting of COVID-19 daily confirmed cases using ARIMA and Prophet. The approach included robust data preprocessing, exploratory analysis, model training, prediction, and evaluation. ARIMA proved to be more accurate due to the data's stationary nature and non-seasonal characteristics during the forecast period. This documentation illustrates how classical and modern forecasting methods can be applied and compared for real-world epidemiological data.

8. References

Hyndman, R.J., & Athanasopoulos, G. (2021). Forecasting: Principles and Practice.
Facebook Prophet Documentation. https://facebook.github.io/prophet/
Statsmodels Documentation. https://www.statsmodels.org/stable/tsa.html
Johns Hopkins CSSE COVID-19 Dataset. https://github.com/CSSEGISandData/COVID-19